# REPORT DOCUMENTATION PAGE

*Form Approved*
*OMB No. 0704-0188*

| 1. AGENCY USE ONLY (Leave blank) | 2. REPORT DATE<br>Aug. '96 | 3. REPORT TYPE AND DATES COVERED<br>Final Report   Nov. '93 – May '96 |
|---|---|---|

| 4. TITLE AND SUBTITLE<br><br>Adaptable Locally-Interconnected Architectures | 5. FUNDING NUMBERS<br><br>N00014-94-1-0090, P0002 |
|---|---|

**6. AUTHOR(S)**

Lex Akers

| 7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)<br><br>Department of Electrical Engineering<br>Arizona State University<br>Tempe, AZ   85287-5706 | 8. PERFORMING ORGANIZATION REPORT NUMBER<br><br>1 |
|---|---|

| 9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)<br><br>Office of Naval Research<br>800 N. Quincy Street<br>Arlington, VA   22217-5660 | 10. SPONSORING/MONITORING AGENCY REPORT NUMBER |
|---|---|

**19960813 157**

**11. SUPPLEMENTARY NOTES**

| 12a. DISTRIBUTION/AVAILABILITY STATEMENT<br><br>Public | 12b. DISTRIBUTION CODE |
|---|---|

DTIC QUALITY INSPECTED 4

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

**13. ABSTRACT (Maximum 200 words)**

This project has built on our previous work in developing a theory of the processing power of locally-interconnected architectures and the studies of the implementation of such architectures. Three locally interconnect architectures, inspired by biology, were developed and fabricated in VLSI. These systems are an analog Gaussian basis circuit integrated with a non-volatile storage memory cell, a localed spatial frequency filter, and a habituation system.

Analog Gaussian basis circuit integrated with a non-volatile storage memory cell was developed. Hardware implementations of the Gaussian basis circuit with on-chip learning is needed for real time and portable applications. Each Gaussian basis cell is symbolically inter-linked with its own long-term storage memory cell forming a highly localized architecture. Experimental results were obtained.

Receptive field structures found in the visual cortex of the mammalian brain act as oriented, localized spatial frequency filters. These receptive field structures resemble Gabor filters. We have implemented analog VLSI cells whose outputs resemble the receptive field profiles.          (Cont.)

| 14. SUBJECT TERMS<br><br>Neural Networks, Analog VLSI | 15. NUMBER OF PAGES<br>32 |
|---|---|
| | 16. PRICE CODE |

| 17. SECURITY CLASSIFICATION OF REPORT<br>Unclassified | 18. SECURITY CLASSIFICATION OF THIS PAGE<br>Unclassified | 19. SECURITY CLASSIFICATION OF ABSTRACT<br>Unclassified | 20. LIMITATION OF ABSTRACT<br>UL |
|---|---|---|---|

This project has build on our previous work in developing a theory of the processing power of locally-interconnected architectures and the studies of the implementation of such architectures. Three locally interconnect architectures, inspired by biology, were developed and fabricated in VLSI. These systems are an analog Gaussian basis circuit integrated with a non-volatile storage memory cell, a localized spatial frequency filter, and a habituation system.

Analog Gaussian basis circuit integrated with a non-volatile storage memory cell was developed. Hardware implementations of the Gaussian basis circuit with on-chip learning is needed for real time and portable applications. Each Gaussian basis cell is symbolically inter- linked with its own long-term storage memory cell forming a highly localized architecture. Experimental results were obtained.

Receptive field structures found in the visual cortex of the mammalian brain act as oriented, localized spatial frequency filters. These receptive field structures resemble Gabor filters. We have implemented analog VLSI cells whose outputs resemble the receptive field profiles.

And lastly, rehabituation is a feature of habituation which allows a system to more rapidly disregard inputs that are not novel. We have implemented habituation, rehabituation and recovery in hardware. Our experimental results demonstrate these responses.

# Adaptable Locally-Interconnected Architectures

## Final Report

By

Lex A. Akers
Department of Electrical Engineering
Arizona State University
Tempe, AZ
85287-5706

August 7 1996

# Adaptable Locally-Interconnected Architectures

Summary

This is the final report for the contract Adaptable Locally-Interconnected Architectures, ONR N00014-94-1-0090, P00002.

Direct hardware implementations of fully interconnected systems are not practical for large numbers of synthetic neurons. The power dissipation resulting from charging and discharging long interconnection lines would dominate the power budget of the chip. This would make many systems either too energy consuming for portable systems, or the special chip packages and system cooling requirements would make the system too expense for mass use. However, a paradigm called cellular automata exists that exhibits massively-parallel processing using only local interconnections. While studies of this processing paradigm have not progressed as rapidly as those on neuromorphic systems, recent work in massively parallel computation has focused on cellular structures. Recently, a merging of these two distinct paradigms has emerged through the concept of cellular neural networks, developed primarily by Chua and his co-workers. Independently of this, the neural research group at Arizona State University (ASU) has pioneered the concepts of locally-interconnected networks from the viewpoint of constraints imposed by VLSI design. Architectures of locally-connected nonlinear analog processing nodes have been demonstrated by our group to have powerful signal processing capabilities.

Many of the currently simulated and electronically implemented neural systems are only feedforward with modal training accomplished as a separate, often off-line, procedure. This restriction is in direct contrast to biological neural systems where processing is local in a spatial-temporal concept, and learning is an integral part of the architecture. The computational principles and cooperative behavior emerging from such spatio-temporal dynamics with learning are largely unknown. However, the development of the cellular architectures discussed above allows us to begin to study the ability of these networks to both compute and learn.

We have developed and demonstrated that virtually large networks can be built up from locally connected Gaussian basis cells. While this has been known mathematically and simulated for a number of years, **we have developed the first practical hardware implementation of a Gaussian basis cell that will**

1

**scale to large number of units.** The unit cell is a low voltage, low power cell that generates a Gaussian roll-off response around a programmable center. Also, this cell has the characteristic of allowing these cells to be interconnected in an efficient, local, and natural way. We also enhanced our previously developed unsupervised categorizing chip to exhibit habituation and rehabituation. Habituation is a biological behavior allowing non-critical information to be disregarded enabling more processing resources for critical tasks. Rehabituation is enhanced disregarding of information previously sampled by the system. We have implemented habituation adaptation in a small temporal mixed signal circuit. This chip demonstrates in hardware the ability of a system to focus on important task and disregard noncritical inputs. Lastly, we have studied neurons with response characteristics that are locally tuned to a particular range of the input variable. Such cells have been found in many parts of the central nervous system. Examples include cells in the somatosensory cortex that respond selectively to stimulation from localized regions of the body surface, and orientation selective cells in the visual cortex that respond selectively to stimulation which are both local in retinal position and local in angle of object orientation. Populations of these locally-tuned cells have been found organized in cortical maps where the input variable varies in an approximate linear fashion with position in the map. These maps are also observed to have overlapping receptive fields. Overlapping receptive fields offer the capability to improving signal to noise ratios and for providing fault tolerance. The transfer function of these cells are a Gaussian convolved with a sin or cos wave, refereed to as a Gabor logon. **We have design and tested the first hardware implementation of the Gabor logon.**

This work has resulted in 12 conference, 2 book chapters and 2 journal papers, with 1 manuscript in review. The published papers are:

1. "A Temporal Neural System", Proceedings of the IEEE International Conference on Circuits and Systems, London, Vol. 6, p. 277 1994,(with J. Heeb).
2. "Pattern Recognition and System Control with a Neural Processor", Proceedings of the IEEE International Conference on Circuits and Systems, London, Vol. 6, p. 257, 1994(with J. Donald).
3. "Dimensional Reduction of Analog Signals with a Neural Processor", Proceedings of the IEEE International Conference on Neural Networks, (invited paper)1994(with J. Donald).

4. "Spatial-Temporal Neural Systems," Annals of Biomedical Engineering, Abstract Supplement for the Biomedical Engineering Society, (invited paper)1994.

5. "Hardware Implementation of Habituation", Proceedings of the IEEE International Conference on Circuits and Systems, Seattle, Vol. 3, p. 2092, 1995(with T. Wang).

6. "Limited Interconnected Circuits and Architectures", Proceedings of the Helsinki Workshop on the Future of Semiconductors", 1995.
Trends in Microelectronics, Ile de Bendor, France, 1995.

7. "A Multi-Dimensional Analog Gaussian Radial Basis Circuit", Proceedings of the 1996 IEEE International Conference on Circuits and Systems, Atlanta, (invited papers)(with L. Theogarajan).

8. "Analog VLSI Implementations of Receptive Field Profiles Found in Visual Cortical Neurons", Proceedings of the 1996 IEEE International Conference on Circuits and Systems, Atlanta, (with L. Theogarajan).

9. "A Habituation, Rehabituation, and Recovery Chip", Proceedings of the 1996 IEEE International Conference on Circuits and Systems, Atlanta, (with V. K. Goru).

10. " Silicon Models of Visual Cortical Processing", Proceedings of the 1996 IEEE International Conference on Neural Networks, Washington D.C. (with L. Theogarajan).

11. "VLSI Implementation of Rehabituation", Proceedings of the 1996 IEEE International Conference on Neural Networks, Washington D.C. (with V. K. Goru).

12. "A Programmable Gaussian Node", Proceedings of the 1996 IEEE International Conference on Neural Networks, Washington D.C. )(with R. Kuppuswamy, L. Theogarajan).

13. "Synthetic Neural Systems in the 1990s", An Introduction to Neural and Electronic Networks, Ed. by S. V. Zornetzer, J. L. Davis, and C. Lau ,Academic Press, New York, second edition, (Contributing author)(revised and updated chapter).1995

14. "An Unsupervised Neural Processor", Silicon Implementation of Pulse Coded Neural Networks, Ed. by M. Zaghloul, J. Meador, and R. Newcomb, Kluwer Academic(Contributing author with J. Donald). 1995

15. "An Electronic Habituation Chip," Special Issue of Circuits, Systems, and Computers 1996.(In Press)(with T. Wang).

16. "A Scaleable Low Voltage Analog Gaussian Radial Basis Circuit", IEEE Trans. on Circuits and Systems, (in press)(with L. Theogarajan).

The attached papers provide the technical details of our accomplishments.

# A Programmable Gaussian Node

Ravishankar Kuppuswamy, Luke Theogarajan and L.A. Akers
Center for Solid State Electronics Research
Arizona State University, Tempe, AZ 85287-5706
email: lakers@asu.edu

## ABSTRACT

Analog Gaussian basis circuit integrated with a non-volatile storage memory cell is described. Hardware implementations of the Gaussian basis circuit with on-chip learning is needed for real time and portable applications. Each Gaussian basis cell is symbiotically inter- linked with its own long-term storage memory cell forming a highly localized architecture. Experimental results of both the Gaussian basis circuit and the memory element is presented. We show simulated results of an application of our cell in a MERAM system.

## I. Introduction

The human brain is comprised of cells with response characteristics that are locally tuned to a particular range of input variables. These cells access information required from memory bodies located in the vicinity. The Gaussian basis function network is an implementation of a powerful system for learning and approximating complex input-output mappings. However, most of these networks are computer simulations. A hardware implementation is needed for high speed and portable systems. We have designed a Gaussian cell[9] which is extremely compact and is integrated with a non-volatile memory element in a hierarchically connected architecture. Each Gaussian cell has a local memory element thereby emulating the brain structure to a large extent. The integrated architecture is highly expandable and scales up to a large number of processing elements. We present both simulations and experimental results of these circuits.

## II. Gaussian Basis Circuit

The Gaussian function implementations in most of the circuits reported[1, 2, 4, 5, 10] are direct mathematical implementations. Neural models are often simplified for analytical tractability, and are not intended to be an accurate representation of its biological counterpart. Therefore we believe that rather than designing a circuit to give an exact Gaussian, a circuit that has the essential properties of the Gaussian will suffice. This is a peak at the desired center and a nonlinear smooth drop on either side as the input moves away from the center. Therefore, we designed a circuit which has a general Gaussian or "Bump" shape.

When two transistors are connected in series, there occurs a self correlation of currents and if the currents have a differential or complementary nature a bump output results[5]. One way of implementing this differential or complementary nature is to use a differential amplifier. An alternate method is to use a device which has a complementary characteristic to the same input voltage. PMOS and NMOS devices have such complementary characteristics. By using this inherently complementary nature we have been able to design a circuit which approximates a Gaussian surface. The circuit is shown in Figure 1. The input to the circuit is $V_{in}-V_c$, where $V_c$ is the center.

In order to use the exponential relationship between the input voltage and output current, we operate our circuit mostly in the subthreshold region of operation. One way to achieve this is to lower the supply voltage such that both devices operate in the subthreshold region. However, this results

in a very small current for the entire input voltage swing. We have developed a method for the circuit to be in the subthreshold voltage region of operation for the tails of the output current, and be in the saturated above threshold voltage region for the peak of the output current. This gives an excellent peak-to-valley ratio, and good current drive. We do this with a non-linear resistor, in our case by a drain connected PMOS transistor. The PMOS load also facilitates the mirroring of the current to the output transistor. The load transistor drops the voltage to the source of the correlating PMOS transistor as a function of the current through the circuit. Thus the voltage seen by the source of the PMOS transistor when it dominates is lower than the supply and after a few tenths of volts forces the circuit into the subthreshold region of operation. The advantage in using the above method is that the dynamic bias variation to the source of the PMOS transistor allows the circuit to operate to a large extent in the subthreshold regime but helps to keep the supply voltage relatively high. The point where the built-in center occurs is not at Vdd/2 as in the case of a the current through a simple inverter, but at a smaller voltage slightly above threshold. This above threshold operation at the peak is beneficial, since the peak of a Gaussian resembles a quadratic and increases the current drive. The output of the circuit is shown in Figure 4. We have also extended this circuit to handle many inputs using a novel current correlator multiplication method[9].
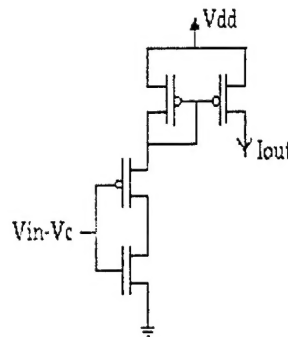


Figure 1. Gaussian basis circuit

## III. Analog Memories

Analog memories for the storage of synaptic weights are generally classified by the duration of charge storage. Short-term storage cells are usually modeled as capacitors[11] at the gate of the transistor and the weight is transferred to the gate using a pass transistor operating as a switch. The leakage current through the junction of this pass transistor limits the storage time. Medium-term storage cells are primarily short-term storage cells with additional refresh circuitry to rejuvenate the charge at periodic intervals of time. Long-term storage elements[6, 7, 11] are usually electrically programmable non-volatile memories where the charge is stored on a floating electrode. The floating electrode is insulated by $SiO_2$, thereby making charge leakage from the gate highly negligible. Our Gaussian basis cell targets applications such as feature recognition and detection. These applications require the weights to be stored for long periods of time and not change on an continuous basis. Our choice of memory naturally targeted the non-volatile storage option. As the cells use hot electron injection and Fowler-Nordheim Tunneling mechanisms for their write and erase procedures, they usually have to be fabricated using special processing techniques. We have hence chosen a storage cell[6, 7] motivated by the Caltech group and fabricated using the $2\mu$ standard MOSIS process to be used for weight storage in the Gaussian basis circuit.
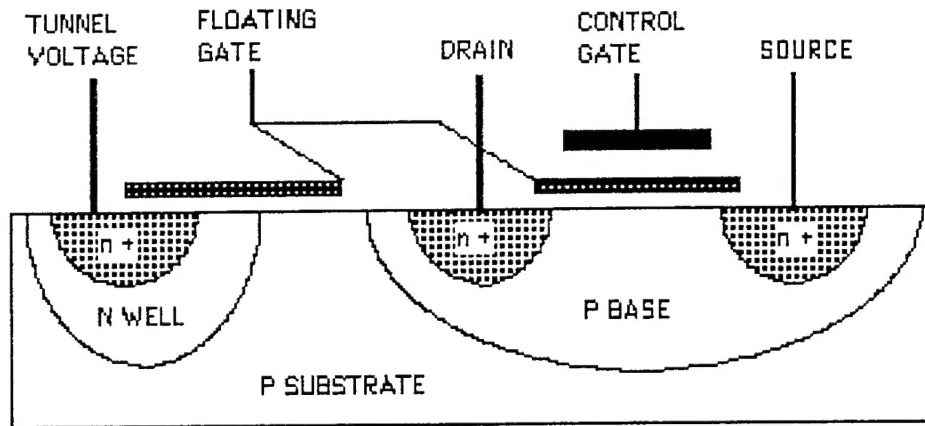
## IV. The Memory Cell



Figure 2. Crossection of non-volatile memory cell. The cell contains p-base layer encompassing the source and drain regions. The four terminals of the cell are source, drain, control gate and tunneling gate

The memory cell[7,8] is basically made of a single NMOS field effect transistor. This transistor is innovatively modified to make it a non-volatile memory storage element fabricated using a standard process. The cross-section of the memory cell is shown in Figure 2. The four basic terminals are the source, drain, control gate and tunneling gate. The source and drain regions are n+ regions lying in a p-substrate, totally enclosed by the moderately doped p-base region. The p-base region plays the pivotal role in the operation of the cell. The p-base layer fulfills two basic functions. Firstly it increases the threshold voltage, thereby creating a region where the potential drops greater than 3.1V within 0.2μm to provide the hot electrons enough energy to cross the oxide barrier. Secondly it also raises the electric field intensity of the channel thereby increasing the rate of hot electron injection. Both these properties act in unison to provide a range of operating drain voltages in the subthreshold region. This cell has a double poly structure with poly1 acting as the floating gate and poly2 as the control gate. The charge is stored on the poly1 floating gate, which is insulated from the rest of the cell by high quality $SiO_2$ making the cell a non-volatile storage element.

The charge on the floating gate is modified using both the tunneling and control gates. The control gate terminal is formed directly on the poly2 layer using a capacitance contact. The poly1 layer runs over a well region and the tunneling gate terminal is formed using a well contact. The charge on the floating gate is modified using both tunneling and hot electron injection mechanisms. A high positive voltage is applied on the well region, attracting the electrons on the floating gate into the well tunneling through the oxide barrier. This tunneling process decreases the electrons on the gate, thereby increasing the voltage on the floating node. Now a large drain voltage combined with a positive voltage on the control gate attracts hot electrons on to the floating gate. This decreases the voltage on the floating electrode.

## V. The $V_{in}$ - $V_c$ Circuit

Most analog VLSI circuits for neural networks employ the differential pair as the heart of the circuit design. This differential pair allows the computation of V1-V2. This design philosophy has its advantages and disadvantages. Our design does not use a differential pair and hence we need a separate circuit to do this computation and is the subject of this section. We have used a long-term memory cell[7] in a novel fashion to store $V_c$. One method of achieving a difference in voltages is to put them in series. This method is however not a very feasible one since most voltages are referenced

to ground. In order to make our circuit compatible to all methods of storage we present a novel two transistor circuit to do the difference computation. The circuit is shown in Figure 3. The drain connected NMOS transistor acts as a resistive load. The voltage at the output node of this circuit is given by

$$V_o = V_{in} - I_d R_d$$

where $I_d$ is the current due to a voltage $V_c$ on the gate of the lower transistor and $R_d$ is the resistance offered by the drain connected NMOS transistor. If the upper and lower transistor are identical then the value of $I_d R_d$ is equal to $V_c$. This achieves the desired computation. For the transistors to be identical the upper transistor should not have any body effect, therefore it should be placed in a separate well. The SPICE output of the Gaussian cell for different $V_c$ voltages using the $V_{in}$-$V_c$ circuit is shown in Figure 5.
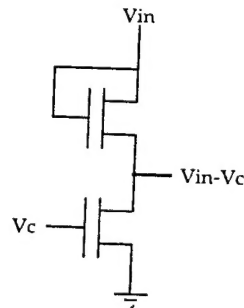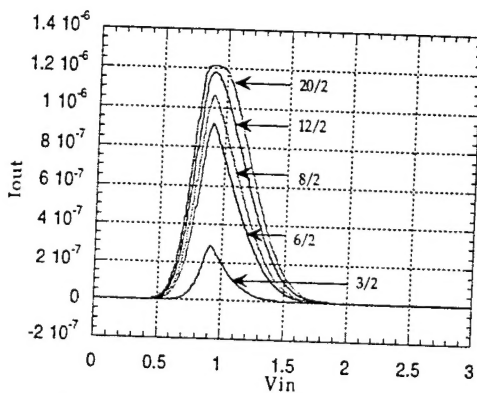


Figure 3. Vin-Vc circuit



Figure 4. The measured output curves of the for equal PMOS and NMOS transistor sizing of 20/2, 3/2 for the largest to the smallest current peaks. The input to the circuit was with $V_c$ equal to zero.
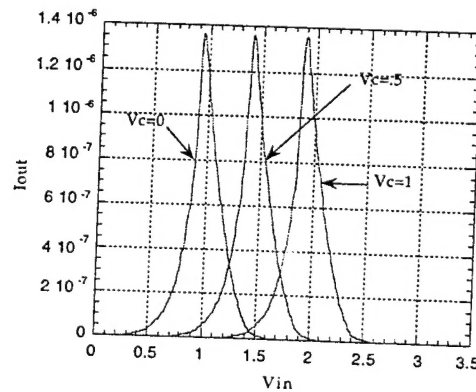
Figure 5. The SPICE simulated output of the circuit Gaussian circuit with the $V_{in}$ - $V_c$ circuit for different $V_c$ voltages

## VI. Memory Testing and Integrated Architecture

The charge on the floating gate represents the weight stored in the memory element. The voltage stored on the floating gate was directly measured by connecting the floating gate to a negative feed-back operational amplifier buffer circuit. A circuit[1] comprising of a NMOS and PMOS transistor with their drains connected to a feedback capacitance was also used. The floating gate was connected to the gate of a PMOS transistor and the capacitance for linear voltage storage control on the other plate of the capacitance. This node can be programmed to store any rail to rail voltage.

The tunneling operation was conducted by applying a high positive voltage on the well (32-35V). This voltage stripped the floating electrode of its electrons and increased the voltage on the gate to about 5-6V. To cease the tunneling operation, the voltage on the tunneling terminal was brought down to 25V and the floating gate voltage was maintained. Accordingly the voltage measured at the capacitance node reads zero after the tunneling operation. Hence the cell is now completely erased. The electrons are now provided with both the energy and direction to cross the oxide barrier. The control voltage is maintained at 8V and the drain voltage is brought up to about 4-5V. This attracts electrons to the floating gate and decreases the voltage on the gate. The capacitance node voltage now increases in a linear fashion. Experimental results indicating voltage changes at the output capacitance node due to both tunneling and injection processes are shown in Figure 6.

The memory cell was combined with the Gaussian basis circuit and different sizes of NxN architectures were designed. The basic idea of the integration process was to efficiently reduce the number of I/O lines. The total area occupied by each memory cell and its read/write circuitry is approximately 70 $\mu m^2$. The memory cell has four basic lines. All the cells of the architecture have common source and control terminals. The drain terminal of each cell was connected to its write circuitry. The tunneling line of all the cells in the architecture were presently independent of each other. On designing larger architectures, a separate addressing scheme would help in optimizing on the number of tunneling lines. The source terminal of all the cells in the architecture was connected to ground. The output weight $V_c$ stored in the memory cell was fed as an input to the $V_{in} - V_c$ computational circuit.
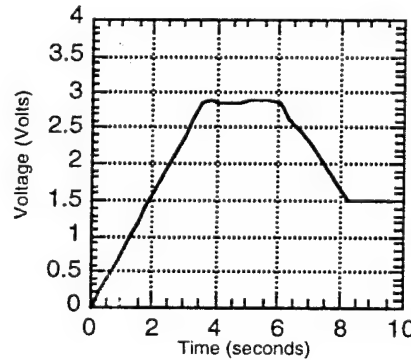


Figure 6. Output voltage of the capacitance during injection and tunneling operations

## VII. Application

The Gaussian basis circuit can be used in Gaussian radial basis classifiers. We show a SPICE simulation of a circuit implementing a multi-valued exponential recurrent associative memory (MERAM) [3]. This application employs a multi-dimensional Gaussian basis circuit as a similarity computing element. Our implementation is a variation of the implementation presented in [8]. A three input, three output, MERAM was simulated. Three patterns of three components were stored. For the first simulation the stored patterns were

$$\begin{pmatrix} 1 & 1 & 1 \\ 1.5 & 1.5 & 1.5 \\ 2 & 2 & 2 \end{pmatrix}$$

The pattern applied to the input of the network was $\begin{pmatrix} 1.5 & 1.7 & 1.2 \end{pmatrix}$. The network settled into the stored pattern $\begin{pmatrix} 1.5 & 1.5 & 1.5 \end{pmatrix}$ as shown in Figure 7.
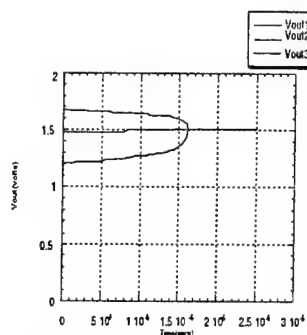
Figure 7. Output of MERAM network node as a function of time

## VIII. Conclusions

We have described a very compact analog Gaussian basis cell and memory element in a highly localized architecture. Both simulations and experimental results demonstrating the operation of these circuits have been presented.

## IX. Acknowledgments

## X. References

[1].    J. Anderson, J. C. Platt, and D. B. Kirk, " An Analog VLSI Chip for Radial Basis Functions", In S. J. Hanson, J. D. Cowan, and C. L. Giles,*Advances in Neural Information Processing Systems,,* Vol. 5, San Maetro, CA; Morgan Kaufmann Publishers Inc., 1993, pp. 765-772.

[2].    S. Churcher, A. F. Murray and H. M. Reekie, "Programmable Analogue VLSI for Radial Basis Function Networks", *Electronics Letters,* 2nd September 1993, Vol. 29, No. 18. pp. 1603-1605.

[3].    T. D. Chieuh and H. K. Tsai, "Multivalued Associative Memories Based on Recurrent Networks", *IEEE Trans. on Neural Networks,* vol4, no. 2, pp. 364-366, Mar 1993.

[4].    Joongho Choi, Bing J. Sheu, and Josephine C.-F. Chang, " A Gaussian Synapse Circuit for Analog VLSI Neural Networks," *IEEE Trans. on VLSI Systems,* Vol.2, March 1994, pp. 129-133.

[5].    T.Delbruck, "Bump Circuits," *Caltech Internal Document,* CNS Memo 26,1993.

[6].    Chris Diorio, Sunit Mahajan, Paul Hasler, Bradley A. Minch and Carver Mead, " A High-Resolution Non-Volatile Analog Memory Cell," *Caltech Internal Document.*

[7].    Paul Hasler, Chris Diorio, Bradley A. Minch and Carver Mead, "Single Transistor Learning Synapses," *Caltech Internal Document.*

[8].    R.J. Huang and T. D. Chieuh, "Circuit Implementation of the Multivalued Exponential Recurrent Associative Memory", *World Congress on Neural Networks,* vol II, 1994, pp. 618-623.

[9].    Luke Theogarajan and L.A. Akers, " A Multi-Dimensional Analog Gaussian Radial Basis Circuit," *Proc. of International Symposium of Circuits and Systems ,* 1996.

[10].   S. S. Watkins and P. M. Chau, "A Radial Basis Function Neurocomputer Implemented with Analog VLSI Circuits", *Proc. IEEE/INNS Int. Joint Conf. Neural Net.,* vol. II, pp. 607-612, Baltimore, MD, 1992.

[11].   Vittoz etal, " Analog Storage Of Adjustable Weights," in *VLSI Design of Neural Networks.* Boston: Kluwer, 1991, pp. 344-363.

# Novel Circuits for Neural Information Processing

Luke Theogarajan and L. A. Akers
Center for Solid State Electronics Research
Arizona State University
Tempe, AZ
85287-5706

*Abstract:* Information processing in the visual cortex of the mammalian brain allows compact image coding and representation. Receptive field structures found in the visual cortex of the mammalian brain resemble Gabor filters. There has been recent interest in the use of such receptive field profiles for image coding and texture processing. Systems employing such Gabor filters have been implemented in software for a variety of applications. We believe a hardware implementation of such cells will be helpful in artificial visual processing. We have implemented analog VLSI cells which perform such Gabor-like functions. We describe experimental results of our circuit. We have implemented both the sine and cosine type of circuits. The sine and cosine circuits described can also be used as edge detection units and on-off center-surround units respectively. Our circuit is the first hardware implementation of a Gabor approximation circuit.

## I. Introduction

Neurons found in the primary visual cortex, i.e. the striate cortex, act as localized spatial frequency filters[1]. The receptive field profiles of such cells resemble even-symmetric or odd-symmetric Gabor filters[1]. There are two classes of orientation-selective cells found in the visual cortex[2]. Cells with two principal subregions, one excitatory and the other inhibitory, and cells with a central on or off region flanked on both sides by an antagonistic surround. The profile of the first class of cells resembles a localized sine wave and the second class a localized cosine wave both localized by a Gaussian-like profile. The first class of cells selectively respond to an edge and the second class responds to bar width. There two schools of thought over the function of the visual cortical neurons. Some researchers believe that the visual cortical neurons act as feature detectors[3], while others believe that cortical neurons act as spatial filters[4,5]. Each view has its own advantages and both

1

views have very strong correlations. A nice argument in favor of feature detection is presented in [6].

Spatial frequency is defined as the change in the luminance as a function of distance. The receptive field structures found in the visual cortex are believed to act as localized spatial frequency filters. In a more mathematical approach they are believed to perform some sort of localized Fourier transform of the incoming signal. A frequency representation of a signal allows a sparser representation of the image than a feature representation[7]. From basic filter theory we know that a signal that is well localized in the time domain has a large frequency spectrum and vice-versa. Gabor[8] proved that a signal which is localized in time by a Gaussian window will have minimum number of components in the frequency domain. This is useful when one wants to perform some localized analysis such as a localized Fourier transform. Fourier analysis can be considered as the projection of the signal onto its set of basis vectors represented by sines and cosines of various frequencies and amplitudes. So a filter in the time domain which resembles a sine or cosine wave of a particular frequency localized by a Gaussian will have minimum joint uncertainty in the time and frequency domain. A series of filters tuned to different frequencies operating on the signal will give an optimal localized frequency representation of the signal. Such filters are known as Gabor filters. J. G. Daugmann[9] extended this principle to the spatial domain and also to two dimensions. In order to extend this form of patchwise Fourier analysis to two dimensional waveforms or patterns we must include the orientation of the pattern. These filters are called the 2D Gabor filters. 2D Gabor filters reduce the joint uncertainty in four dimensions. Gabor filters in the two dimensional spatial domain is mathematically represented by a complex exponential grating localized by a bivariate Gaussian.

A. Spatial frequency versus spatial feature representation

The biological reasons for spatial frequency rather than feature representation is that there is a bottleneck between the optical system and the optic nerve[7](roughly 130 million receptors to 1 million optic nerve fibers). Visual information is efficiently condensed when represented in the spatial frequency domain. This is also true for image processing systems, where the

2

pixel by pixel representation generates a large amount of data which must be efficiently transferred through the communication system utilizing limited bandwidth. Another reason for utilizing spatial frequency representation is that the visual system needs to separate the relatively small luminance variations from the main illuminant. This could be easily accomplished by filtering out low spatial frequency components. Also the number of units needed to encode spatial frequency information is less since we could have units tuned to reasonably broad range of spatial frequencies. If the visual stimuli are spatially periodic then there is a clear advantage of spatial frequency filtering to feature detection, since the signal can be encoded by just its frequency, phase and orientation versus a lot of feature detection units. Even for relatively low spatial periodicities there is an advantage to using the frequency representation. Most objects in the visual world are spatially periodic[7]. Texture processing which is related to the phase information is easily accomplished using the frequency representation.

B. Localized versus Global analysis

A localized analysis is helpful in identifying the slant and depth of textured visual objects[7]. A global Fourier transform would change its frequency spectrum for changes anywhere in the field. However, for many visual problems most areas remain unchanging therefore using a local analysis will allow the system to concentrate on the part of the visual world that is changing. Also visual information is locally periodic in space, and objects in the visual world are mostly spatially compact i.e. all the parts that make up the object tend to be adjacent to each other. These are powerful reasons to use local information processing for visual problems.

In the second section we describe our sine, cosine, and Gabor analog VLSI circuits. Finally, in the third section we summary our contribution.

II. Analog VLSI Implementations

Our circuits model the receptive field profiles found in visual cortical neurons[10]. We have designed these circuits to respond to single points in

3

space which in case of the logon circuits are represented as voltage pairs. We realize that a transformation from space to voltage is necessary.

A. Sine Approximation Circuit

The circuit implementation of a temporal sine wave is well known and is now considered trivial. The same argument does not hold for a sine wave with respect to an independent variable such as an input voltage. There have been two earlier implementations, one using BJTs [11] and the other using MOS transistors[12]. Both the implementations are similar and use the differential current representation. The normalized output results in a sine wave like profile. We have designed a sine approximation circuit. Our circuit closely follows the implementation given in [12].

We have designed the circuit taking many clues from biological systems. In experimentally derived response profiles of the cells in the visual cortex of the cat the sine profile is obtained by subtracting the response of the inhibition cells from the excitation cells[10]. The profile is equivaler.t to a Gaussian profile which is split at its center. Such a profile can be implemented in analog VLSI in a similar fashion. The generation of a Gaussian profile is done by using the circuit implementation of Delbruck[13] as shown in Fig. 1. The circuit is based on a simple current correlator shown in Fig. 2. For correlating transistors Delbruck defined a parameter S by:

$$S = \frac{(W/L)_{middle}}{(W/L)_{outer}} \tag{1}$$

Assuming the top transistor in the stack is in saturation(subthreshold), the output current of the current correlator can be computed as:
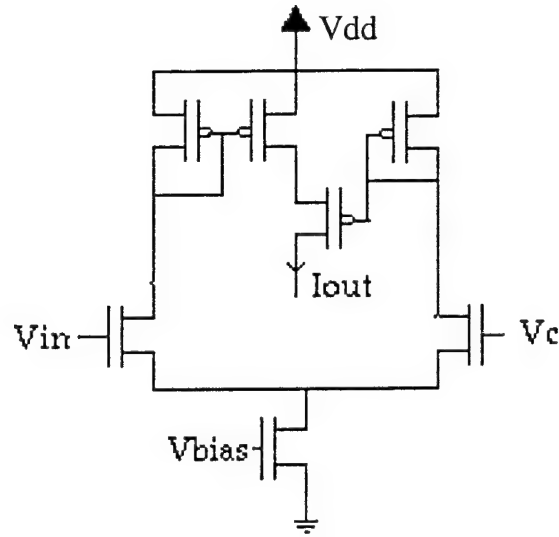
$$I_{out} = S\frac{I_1 I_2}{I_1 + I_2} \tag{2}$$

4

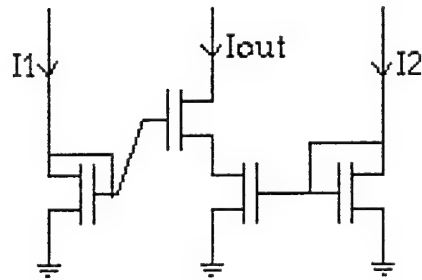Fig. 1: The bump circuit of Delbruck.



Fig. 2: The simple current correlator.

If the input currents to the current correlator are the limb currents of a differential amplifier then substituting this in the current correlator relationship provides:

$$I_{out} = I_b \frac{S}{4} \mathrm{sec}h^2\left(\frac{\kappa\Delta V}{2}\right) \qquad (3)$$

This function gives a bell shaped curve which resembles a Gaussian. We have also reported a circuit that gives a Gaussian profile[14]. However, we will use Delbruck's circuit in the sine implementation because the design

5

allows us to easily generate the sine-like profile. We use our circuit to achieve the Gabor logon profile discussed later.

The output of the circuit in Fig. 1 is fed as a bias current to a differential amplifier. If the input to the differential amplifier is similar to the input of the current correlator then the currents in each limb of the differential amplifier will resemble one half of a Gaussian wave. The input is given by mirroring the current in the input limbs to the limbs of the differential amplifier by means of an additional diode connected transistor as shown in Fig. 3. The circuit implements the following equation:

$$I_{out} = I_b \sec h^2 \left( \frac{\kappa \Delta V}{2} \right) \tanh \left( \kappa^2 \Delta V \right) + I_{dc} \tag{4}$$

where $\Delta V = V_1 - V_2$ and $I_{dc}$ is the dc offset current.

The complete circuit diagram is shown in Fig. 3. Transistors m1-m7 form a PMOS version of the bump circuit discussed earlier. Vc determines the zero crossing of the sine circuit. Vbias is the bias voltage to the circuit. This bias must be sufficiently high in order to supply sufficient current to the diode connected transistors m6-m9. This is necessary to generate sufficient voltage levels to the tanh circuit formed by transistors m10-m13 because the sources of the differential pair transistors m10-m11 are not at ground potential but at some higher voltage.

The output from a fabricated chip is shown in Fig 4. The chips were fabricated in 2μ n-well process using the MOSIS fabrication facility. The data was obtained using an Oscilloscope with disk storage capability. The current from the chip was converted to an voltage by using a npn-BJT emitter follower off-chip. The data recorded from the oscilloscope was graphed using Matlab and Kaleidograph software. The effect of different biasing levels is shown. If the biasing level is slowly increased the bump output becomes more wide, quadratic, and then finally flat. For proper circuit operation the bias voltage needs to be at least a few hundred millivolts above threshold. Transistors m15-m16 set a dc offset to the circuit in order for the circuit to be able to vary in the positive and negative direction. There is evidence in biology that there is a baseline firing rate corresponding to the dc level[15]. The input to these transistors are held at Vdd/2 for proper operation.

6

Transistor m14 converts the voltage representation into a current representation.
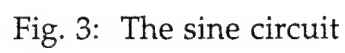
B. Sine Logon circuit

The sine logon is obtained by modulating a sine grating by a bivariate Gaussian. We, however, generate the sine logon profile by modulating the output of a edge detection unit by a Gaussian whose input is orthogonal to the input of the edge detection unit. Both profiles resemble each other. The circuit implementing the sine logon is shown in Fig. 5. The bias of the sine circuit shown in Fig. 3 is replaced by the Gaussian circuit presented in [14] and is shown in Fig. 6. The PMOS and NMOS transistors in series act as complementary current correlators. The drain connected PMOS load serves to keep the circuit in the subthreshold region of operation for a large part of the input and also facilitates the mirroring of the current. For symmetrical operation Vdd is kept at 3V. A more detailed description is given in [14]. The advantages of using this circuit are it supplies higher currents. The need for a higher current occurs because the diode connected transistors m7 and m8 need a sufficient input current to generate voltage ranges which allow the differential amplifier to operate properly. Another advantage is that this circuit can be easily made multi-dimensional as shown in [14]. This allows the input to be a complex feature such as lines. We however show results only from the simple implementation. The bias current effectively controls the amplitude of the sine wave and when modulated by a Gaussian bias current results in a sine logon-like profile. The measured output current from this circuit is shown in Fig. 7.

C. Cosine approximation circuit

Our cosine approximation circuit implements an on-center off-surround profile which resembles a Gaussian localized cosine wave. A circuit for generating a cosine wave using BJTs is given in [11]. We, however, use our own implementation using MOS transistors. Our implementation uses a fewer number of transistors.

Our circuit is shown in Fig. 8. Transistors m1-m7 implement the bump circuit discussed earlier. The output of the bump circuit is fed as a bias

current to the differential amplifier. Also the bump circuit output is fed as an input to the differential amplifier. If the other input to the differential amplifier is a voltage that lies in the near the tail of the bump circuit then a cosine like output results. Since we do not know this voltage *a priori* a simple bias will not suffice. We now consider the design of such a bias. Intuitively we know that the peak current occurs when the input to the differential amplifier are equal. Thus if we duplicate this circuit and keeping the inputs equal to the center voltage we can obtain a current that is equal to the current at the peak. But we need the current to be slightly smaller, therefore we have to make the S ratio of the bias circuit slightly smaller than the S ratio of the bump circuit. We used S ratios of .5 for the bump circuit and .48 for the bias circuit. This ratio is calculated assuming equal currents in both limbs to simplify the calculation. This elaborate bias is done to provide robustness to fabrication parameters. Transistors m8-m14 forms the bias circuit, m15, m16, m20 and m21 are the diode connected mirroring transistors. Transistors m17-m19 and m22-m23 form the differential amplifier, m24 converts the voltage to current representation. Transistors m25-m26 forms the dc current bias to the circuit. The measured output of this circuit is shown in Fig. 9. The transistor sizes for the cosine approximation circuit are shown in Table 1.
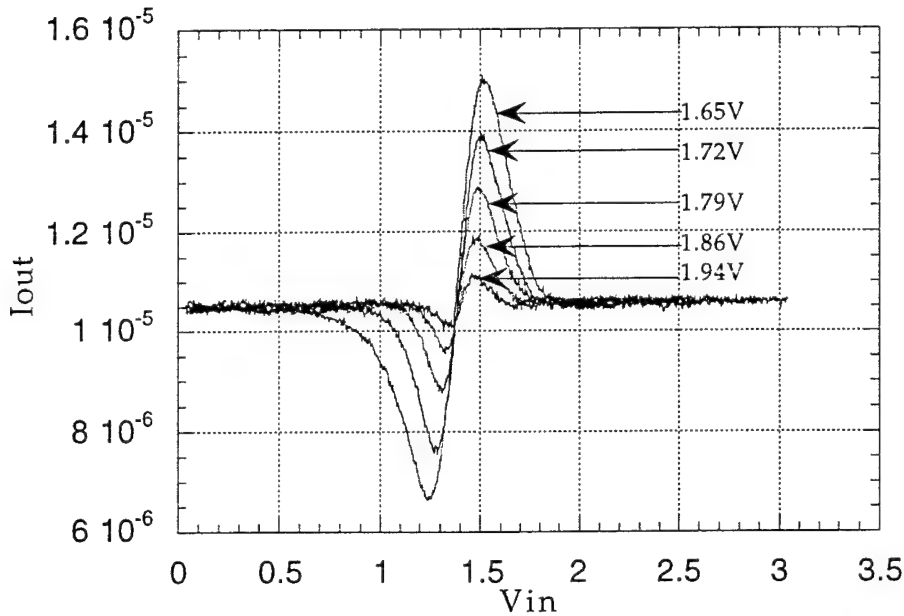
Fig. 3: The sine circuit

Fig. 4: The measured output currents for different bias voltages to the Sine approximation circuit. Vc was held at 1.4 volts.

## D. Cosine Logon circuit

The cosine logon is designed using the same design philosophy as the sine logon circuit. The complete circuit is shown in Fig. 10. The measured output of the circuit is shown in Fig. 11.

In addition to the above described functioning of these circuits we can change the output waveform by changing the voltage on the Vdc inputs. If Vdc is below Vdd/2 then the negative or off lobe of the output is decreased and the positive lobe dominates. If Vdc is greater than Vdd/2 then the negative lobe will dominate. There are many receptive fields with asymmetric lobes in the visual cortex of the cat[10]. Our circuit can be used to model a variety of receptive field by effectively controlling the Vdc input. We can also obtain profiles that are inverted to the ones shown in Fig. 4 and Fig. 9. This can be done by subtracting the limb currents in an opposite fashion to the ones shown in Fig. 3 and Fig. 8.
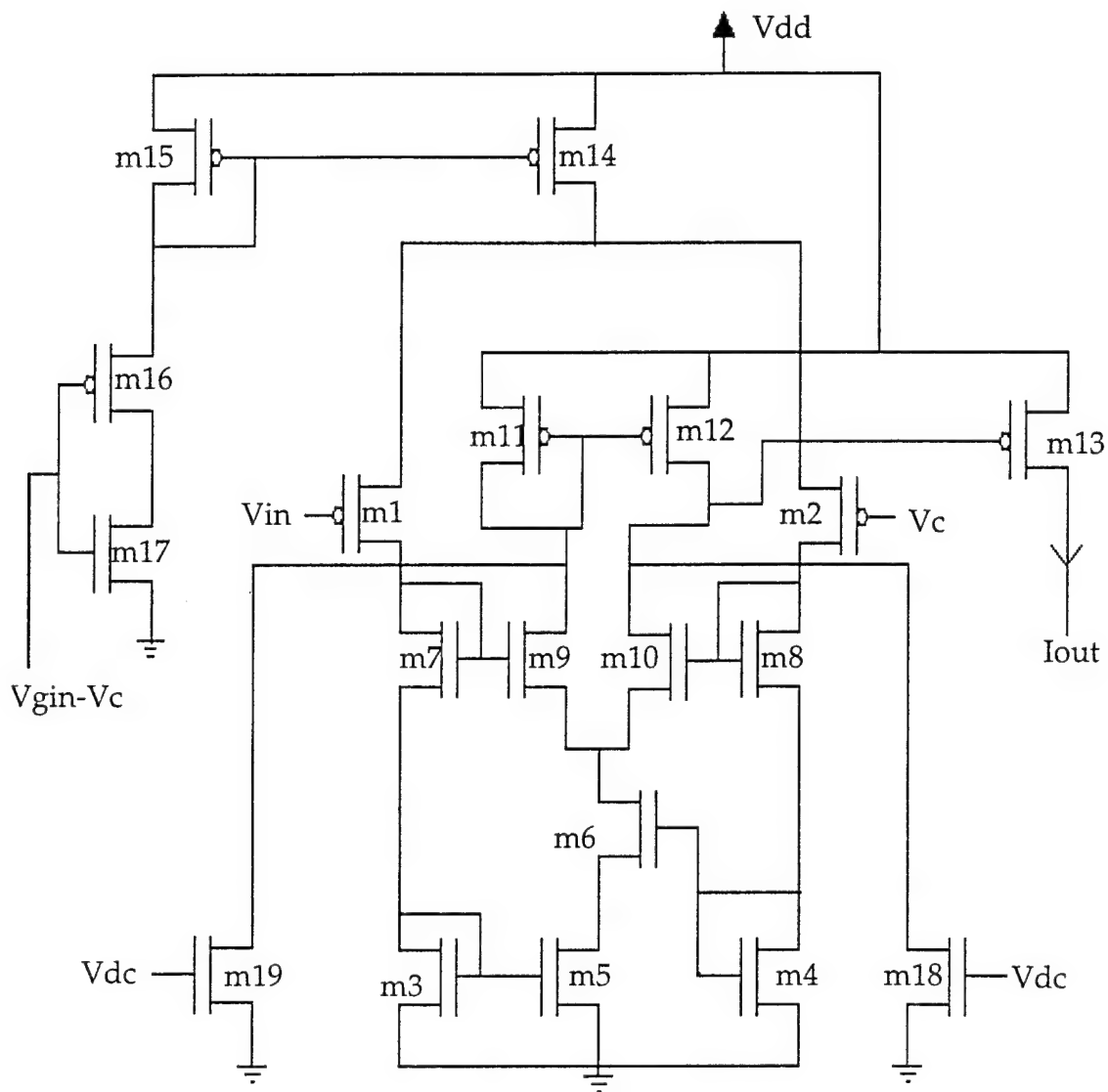
10

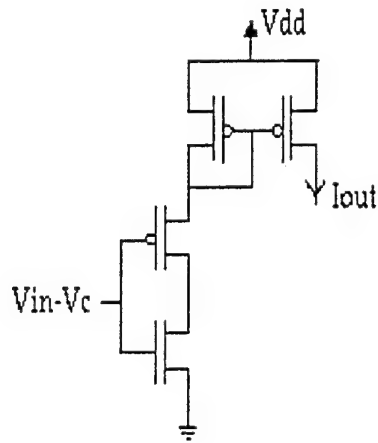Fig. 5: The Sine Logon approximation circuit.

Fig. 6: Our Gaussian cell

III. Conclusion

Visual cortical neurons perform both as localized spatial frequency filters and simple feature detectors. We have presented compact analog VLSI circuits which perform similar operations. We believe that such cells will be very useful in artificial visual processing. The cells presented may be used as Gabor filters or as simple on-center off-surround cells. The cosine circuit may be additionally used as the Mexican hat activation function in neural networks.
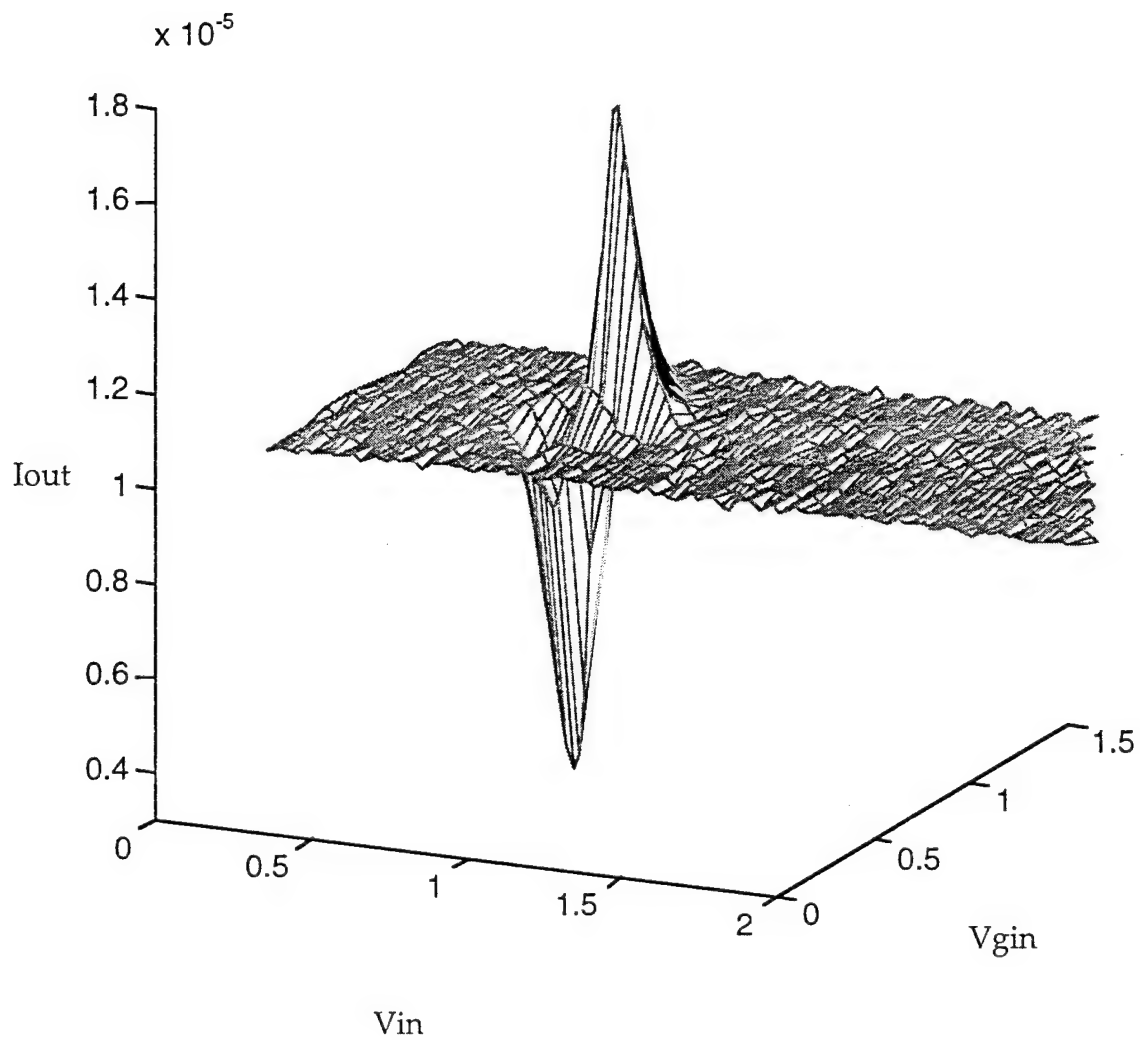
12

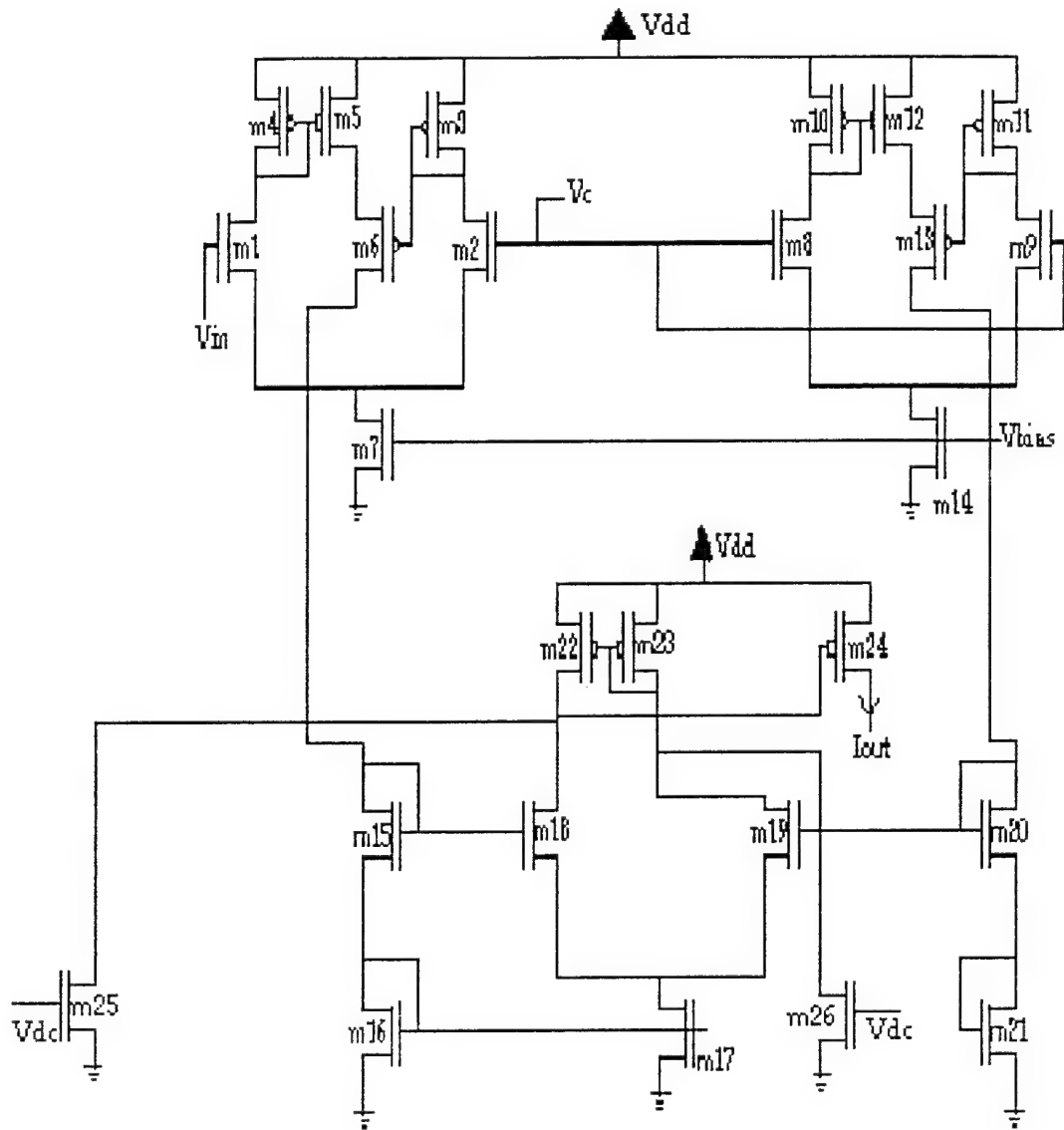Fig. 7: The measured output of the sine logon circuit.

Fig. 8: The Cosine approximation circuit

| Transistor | W/L |
|------------|-----|
| M1 - M12 | 3/2 |
| M13 | 7/5 |
| M14 - M23 | 3/2 |
| M24 | 4/2 |
| M25, M26 | 3/2 |

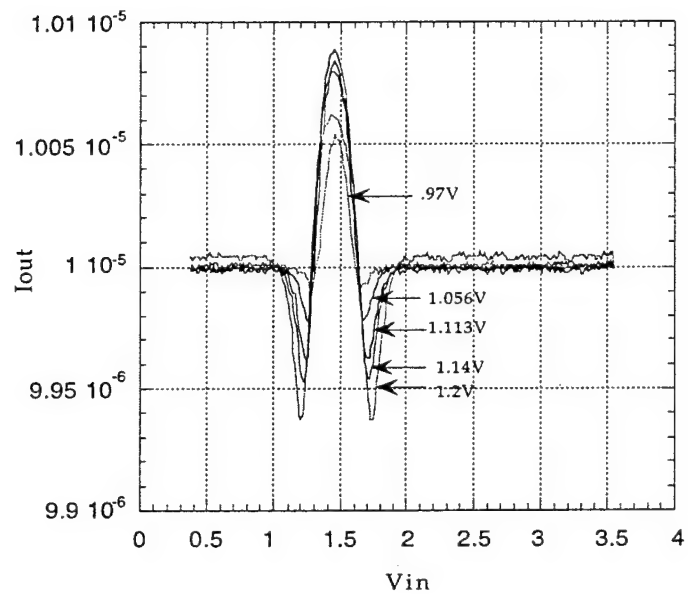Table 1: Transistor sizes for the Cosine approximation circuit.

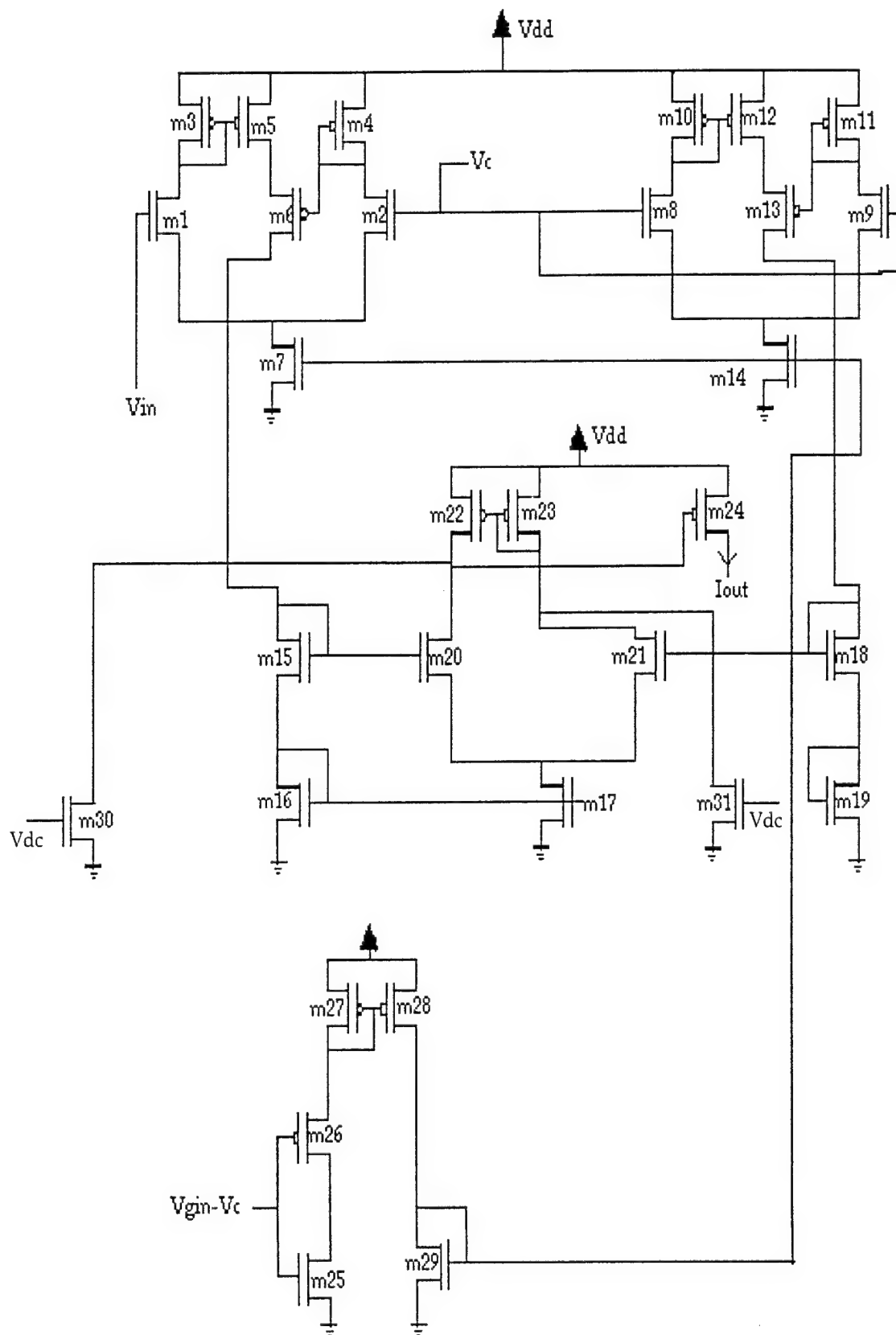Fig. 9: The measured output of the cosine circuit.

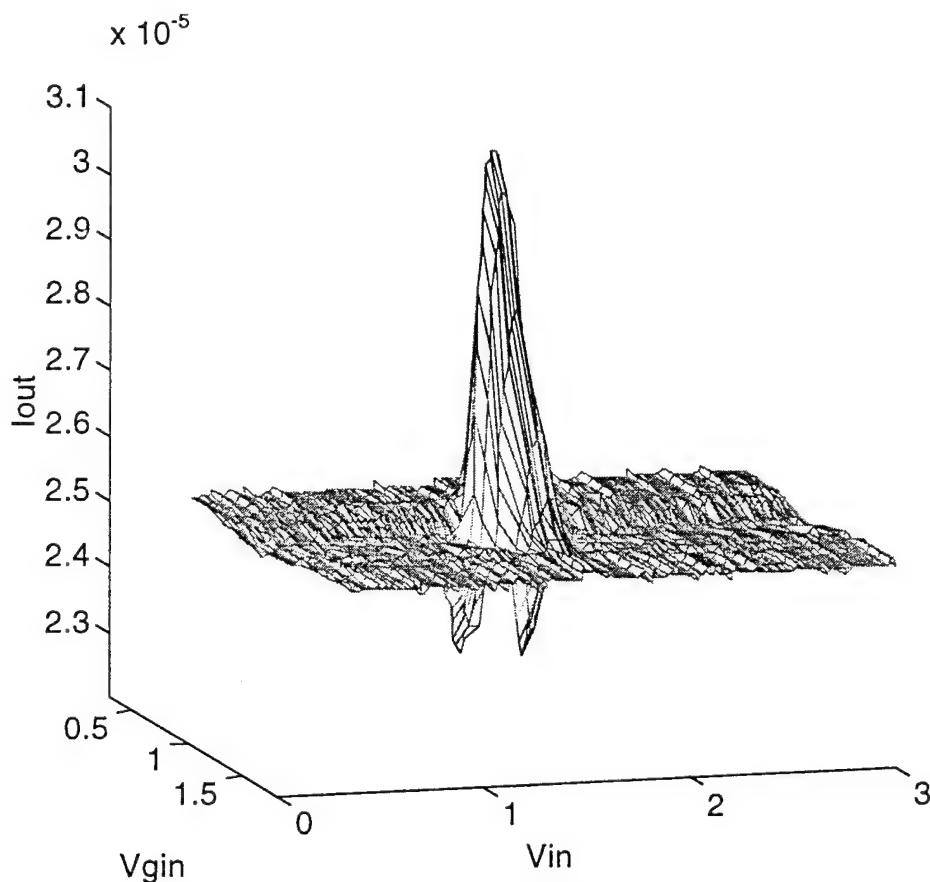Fig. 10: The Cosine logon Approximation circuit.

Fig 11. The measured output of the cosine logon circuit.

IV. References

1. D. A. Pollen and S. F. Ronner, "Visual Cortical Neurons as Localized Spatial Frequency Filters", *IEEE Trans. on Systems, Man and Cybernetics*, Sept/Oct. 1983.

2. D. H. Hubel and T. N. Wiesel, "Receptive fields, binocular interaction and functional architectures in the cats visual cortex", *Journal of Physiology*, Vol. 160, pp. 106-154, 1962

3. P. Mueller, D. Blackman and R. Furman, "Neural Computation of Visual Images", In *An Introduction to Neural and Electronic Networks*, Steven F. Zornetzer, Joel L. Davis, Clifford Lau, editors, Academic Press, 1990.

4. D. A. Pollen, J. R. Lee and J. H. Taylor, "How does the Striate Cortex Begin Reconstruction of the Visual World", *Science*, vol. 173, pp. 74-77, July 1971.

5. L. Maffei and A. Fiorentini, "The Visual Cortex as a Spatial Frequency Analyzer", *Vision Research* Vol. 13, pp. 1255-1267. Pergamon Press 1973.

6. D. M. MacKay, "Strife over visual cortical function", *Nature*, vol. 289 , pp. 117-118., 15 Jan. 1981

7. R. DeValois and K. DeValois, Spatial Vision, Oxford Press 1980

8. D. Gabor, " Theory of Communication", IRE proceedings 1946.

9. J. G. Daugman, "Complete discrete 2D Gabor transforms by neural networks for image analysis and compression", *IEEE Trans. on Acoustics, Speech and Signal Processing*, 36, pp. 1169-1178, 1988.

10. J. P. Jones and L. A. Palmer, "An evaluation of the two dimensional Gabor filter model of simple receptive fields in cat striate cortex", *Journal of Neurophysiology*, 18, 229-289, 1987.

11. E. Seevinck, Analysis and Synthesis of Translinear Integrated Circuits, *Studies in Electrical and Electronic Engineering*, Elseiver, 1988.

12. R. G. Benson, *Ph.D. Thesis*, California Institute of Technology, 1993.

13. T. Delbruck, "Bump Circuits", *Caltech Internal Document*, memo 26, 1993.

14. L. Theogarajan and L. A. Akers, "A Multi-dimensional Analog Gaussian Radial Basis Circuit", International Symposium of Circuits and Systems, (in press)1996.

15. D. S. Touretzky, A. D. Redish, H. S. Wan, "Neural Representation of Space Using Sinusoidal Arrays", *Neural Computation*, 1993.

# VLSI Implementation of Rehabituation

Vijay K. Goru and L. A. Akers
Center for Solid State Electronics Research
Arizona State University
Tempe, AZ
85287-5706
lakers@asu.edu

## ABSTRACT

Rehabituation is a feature of habituation which allows a system to more rapidly disregard inputs that are not novel. We have implemented habituation, rehabituation and recovery in hardware. Our simulations and experimental results demonstrate these responses.

## I. Introduction

The phenomenon of habituation, the waning of responsiveness to repeated or constant stimulation is an important aspect of biological neural network operation. In this paper, we modify our earlier architecture[8,9] to incorporate rehabituation, an enhanced habituation process found in biological systems[7]. By including rehabituation, a higher rate of habituation can be maintained to indicate more familiarity with the input and a lower rate of habituation to less familiar signals. Our goal is to design a system implementing three biological processes: habituation, rehabituation and recovery.

## II. Biological Motivation

Extensive studies have been done on a marine gastropod *Aplysia*[4]. We did experiments on a marine annelid polychaeta of the genus *Sabellastarte Magnifica*, a sedentary tube dwelling worm. These segmented worms build tubes of sand cemented with mucus. The large worms feed by filtering the water for suspended particles with its crown of tentacles called radioles surrounding a central mouth. The tentacles are hollow and muscular and capable of a variety of twisting movements and is the only organ for food and defense.

When a potentially noxious tactile stimulation was given to this worm it was observed to withdrawal its tentacles. The tentacle withdrawal response decreases as a function of increased repetition of an unchanged stimulus. This is called habituation. It was also found that habituation was faster when it remembered its previous experience. This is called rehabituation. This experiment demonstrates that there is a clear cut discrimination of novel from familiar objects. Habituated response was restored in magnitude by withholding stimulation. This is called spontaneous recovery. We model these three processes: habituation; rehabituation; and spontaneous recovery in our chip.

The tentacle withdrawal response is achieved by an impulse to a fast gaint fiber system which contracts longitudinally from end to end[1]. A worm which fails to habituate its reflexive withdrawal response would run the risk of starvation, suffocation and death. Habituation is a biologically adaptive technique to reconcile the need to eat against the threat of being eaten.

## III. Hardware Implementation

Biological systems use habituation to determine detection of predators and rapid escape. Electronic systems can benefit utilizing these processes for allowing non-critical information to

be disregarded enabling more processing resources for critical tasks in applications like detecting stuck-at-1 faults, intelligent door opening systems etc.,

The floorplan of an habituation, rehabituation and recovery system is shown in Fig. 1. The first block is a fully connected unsupervised Oja's classifier[6]. An analog signal is presented as input to the classifier. Oja's classifier assigns the input into one of three classes. The next section of the chip performs habituation, rehabituation and recovery depending upon the repetitiveness of the input. The weight selection block picks the proper weight depending on Oja's classifier and passes this weight to the weight to frequency converter output block which generates a frequency modulated output.
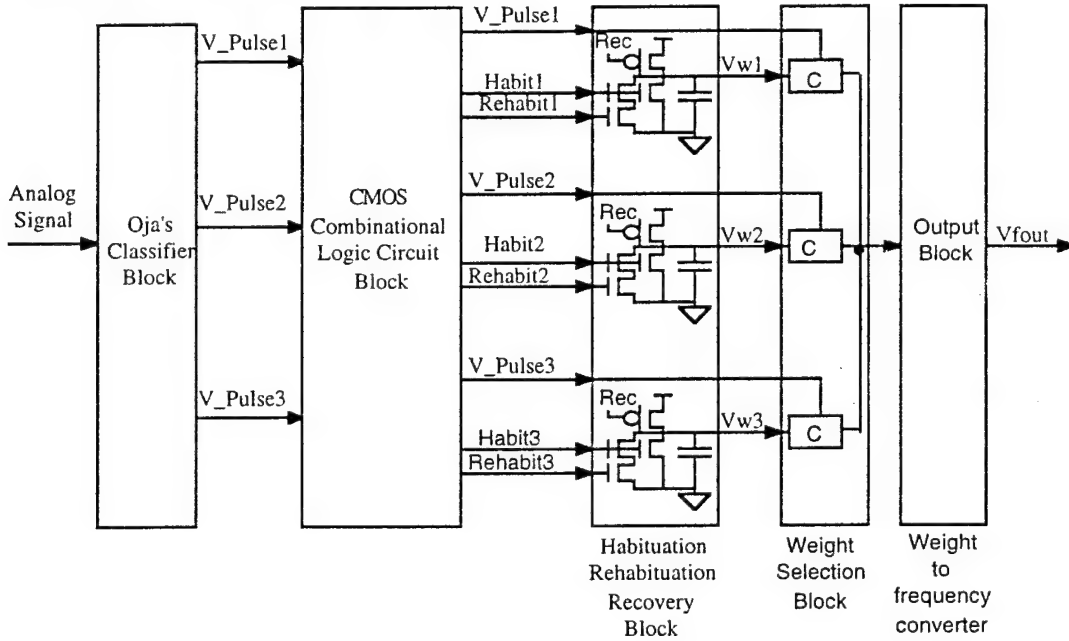


Fig. 1: Floorplan of habituation, rehabituation and recovery system

We have previously reported a hardware implementation of the Oja's classifier in Ref. [2,3]. A competitive learning algorithm is implemented so that only one neuron will fire at any time. While this chip has been built and is operational, for these experiments we have implemented these functions in software so we can vary the duty cycle of this pulse.

CMOS combinational circuit block generates Habit and Rehabit signals depending upon the history of the classification. This circuit considers a classified signal as a novel signal when it no longer remembers this classification. The circuit contains CMOS delay elements which store the classification. The output pulse from the Oja's classifier is fed to a series of four memory elements operating in synchronous mode. The clock is generated by an OR function which is a sum function of all classified pulses at any time. The clock is generated whenever any neuron in the Oja's classifier pulses. Habit signal is generated by an OR function of generated signals and Rehabit is the sum of products of generated signals.

The weight in each neuron in the habituation, rehabituation and recovery circuit represents the responsive ability of the neuron to its corresponding input signal pattern. The weight modification depends upon the number of times the input pattern repeats. The weight modification is governed by

$$\frac{dw}{dt} = R_{rec} * f_{rec\_pulse} - R_{habit} * f_{habit\_pulse} - R_{rehabit} * f_{rehabit\_pulse} \qquad (1)$$

where $R_{rec}$, $R_{habit}$ and $R_{rehabit}$ are the adjustable recovery, habituation and rehabituation rates controlled by external bias signals. The $f_{rec\_pulse}$ is the neural restoring frequency. The $f_{habit\_pulse}$ and $f_{rehabit\_pulse}$ frequencies of habituation and rehabituation which are generated by the CMOS combinational logic block depend on the repetitiveness of the input patterns. We assume that each habituation pulse has the same impact on the weight. Similarly we assume that each rehabituation pulse has equal impact on the weight but in value different than the habituation impact. The only changing parameters in Eq.(1) are the $f_{habit\_pulse}$ and $f_{rehabit\_pulse}$ which depend on the corresponding input signal. Figure 2 shows the weight habituation, rehabituation and recovery circuit.

The initial voltage of the weight capacitor, $V_W$, in each neuron is 3.3V. The weight voltage represents the neuron's responsive ability and is used to control the bias current in the output block if this neuron is selected. When $V_W$ is 3.3V and selected, the output block which contains the weight to frequency converter has enough bias current to generate a maximum output response in terms of frequency modulated output. When Vw is less than 3.3V, the smaller bias current will cause a smaller response and hence shows the habituation effect. The weight voltage $V_W$ is modified by the discharging current $I_{habit}$ and $I_{rehabit}$ and charging current $I_{rec}$. Bias voltage $V_{habit\_rate}$, $V_{rehabit\_rate}$ and $V_{rec\_rate}$ control the magnitudes of $I_{habit}$, $I_{rehabit}$ and $I_{rec}$. Consequently they determine the habituation, rehabituation and recovery rates respectively. The bias voltages are externally controlled such that T5 and T9 are operated near the subthreshold region so that the power consumption is low. During habituation, the only discharge path to the ground is provided by the Vhabit and during the rehabituation period, discharge paths are provided both by Vhabit and Vrehabit so that rapid discharge of the capacitor is accomplished producing a faster habituation.
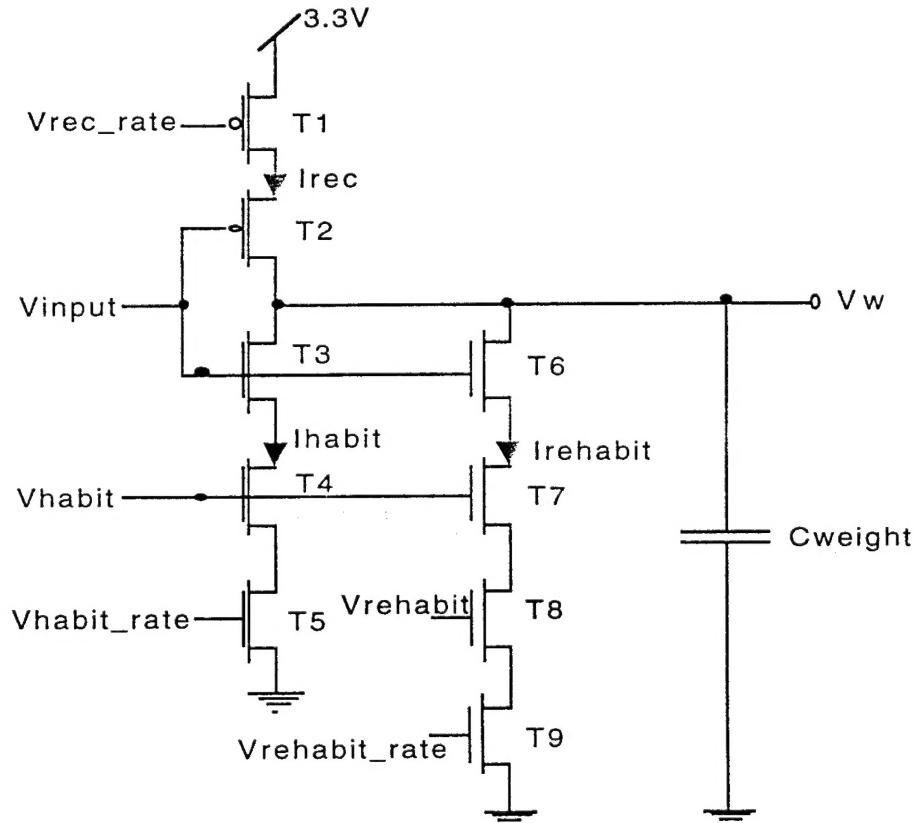


Fig. 2: Weight habituation, rehabituation and recovery circuit in one of the neurons.

There are three neurons in our system and each neuron has a weight representing the responsive ability to one signal. The classified output pulse V_pulse and its complementary value V_pulse*, in each neuron is used to control the transmission gate. The weight corresponding to the neuron which fires the pulse is selected.

We adopt the pulsed self-depleting output circuit from Mead[5] and Donald and Akers[2,3] to generate the output pulses. We generate frequency modulated output pulses. The frequency is proportional to the weight voltage. To reduce power dissipation we cut the output off during the recovery period.

## IV. Experimental Results

This Rehabituation chip was fabricated by MOSIS using a 2 micron n-well process. The chip has one input signal, three classifier pulse inputs, four bias voltages and two outputs. The classifier was implemented in a C program. Output of the chip consists of frequency modulated output and selected analog weight. Figure 3 shows the experimental setup to test the chip. We generate frequency modulated output pulses. The frequency is proportional to the weight voltage. To reduce power dissipation we cut the output off during the recovery period. The output pulse duty cycle depends on the values of storage capacitors. For an unhabituated output, the output frequency has been set to 250kHz and the frequency decreases as the values of selected weight decreases. Figure 4 shows extracted and simulated output results from our chip showing the frequency modulated output voltage and the weight voltage. The weight voltage for habituation is the first breakpoint, and the weight voltage for rehabituation is the second breakpoint. Figure 5 shows the experimental results of input pulses and the habituation, rehabituation and recovery of weight voltage. We used an external capacitor of 20nF to allow a slower output frequency and better observability with our data acquisition system during measurements.
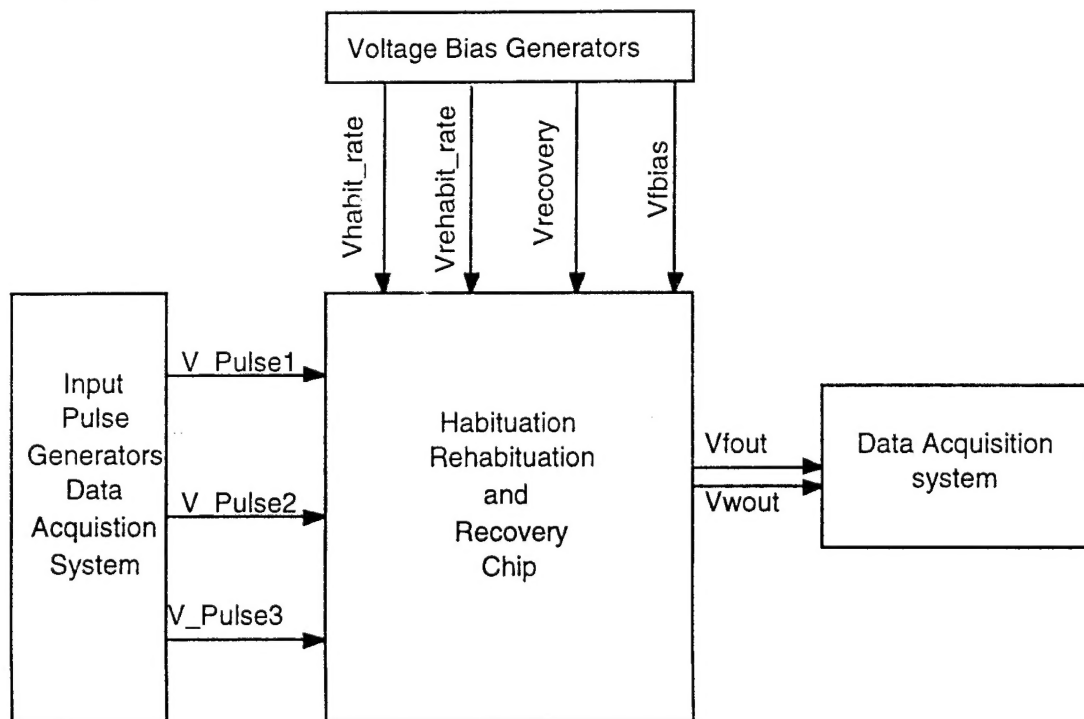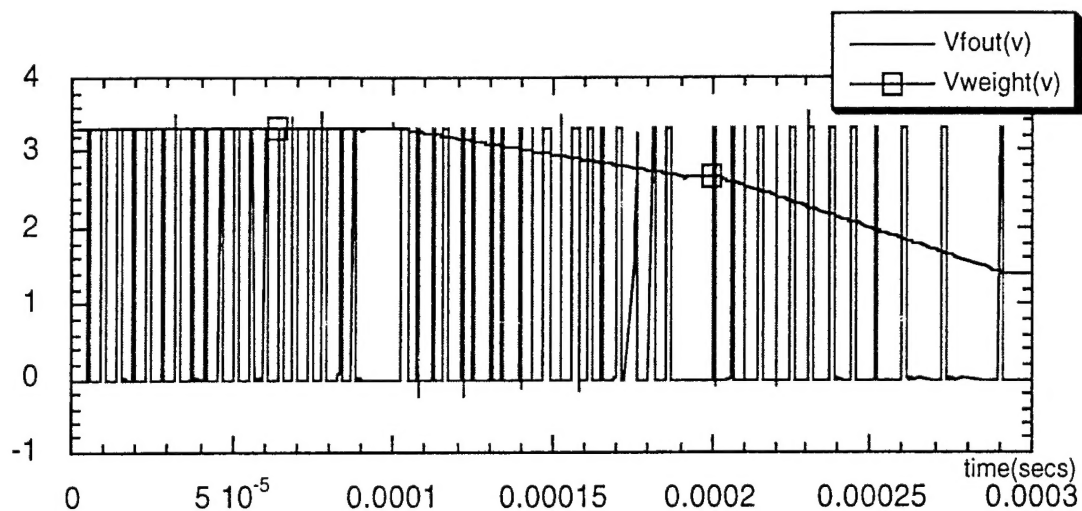
Fig 3: Experimental setup

Fig. 4: Simulated output results from our chip showing habituation and rehabituation.
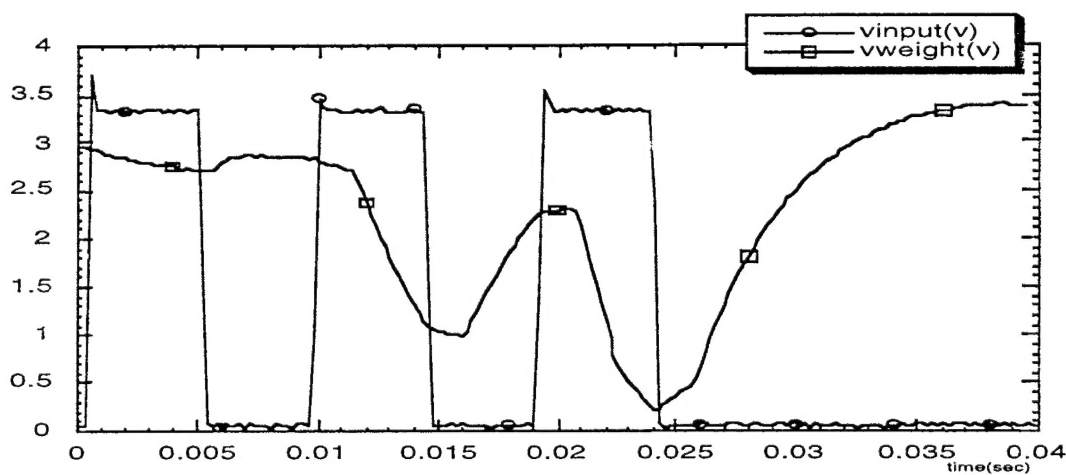


Fig. 5: Experimental results

## V.  Conclusion

We have modified our earlier circuit implementation of habituation and recovery to handle rehabituation. We demonstrated how rehabituation enhances habituation for repeated signals in circuit implementation.

## VI.  References

[1]  Corning W. C., Dyal J. A., and Willows A. O. D., *Invertebrate Learning*, Protozoans Through Annelids, Vol. 1, New York, Plenum Press, 1973.

[2]     Donald J., and Akers L. A., "A Neural Processing Node with On Chip Learning", *Proceedings of the IEEE Internatonal Conference on Circuits and Systems*, Chicago, p. 2748, 1993.

[3]     Donald James and Akers L. A., "An Adaptive Neural Processing Node", *IEEE Transaction on Neural Networks*, Vol. 4, pp. 413-426, May 1993.

[4]     Kandel E. R., and Scwartz J. H., Eds., *Principles of Neural Science.* New York: Elsevier, 2nd Edition, 1985.

[5]     Mead C., *Anlaog VLSI and Neural Systems.* Reading, MA:Addition-Wesley, 1989.

[6]     Oja E., "A Simplified Neuron Model as a Principal Component Analyzer", *Journal of Mathematical Biology*, Vol. 15, pp. 267-273, 1982.

[7]     Peeke H. V., and Herz M. J., *Habituation*, Vol. 2, Academic Press: New York and London, 1973.

[8]     Wang T. and Akers, L. A.,"Hardware Implementation of Habituation", *Proceedings of the IEEE International Conference on Circuits and Systems*, Seattle, Vol. 3, p. 2092, 1995.

[9]     Wang T. and Akers, L. A., "An Electronic Habituation Chip", Special Issue of *Circuits, Systems and Computers*, 1995.